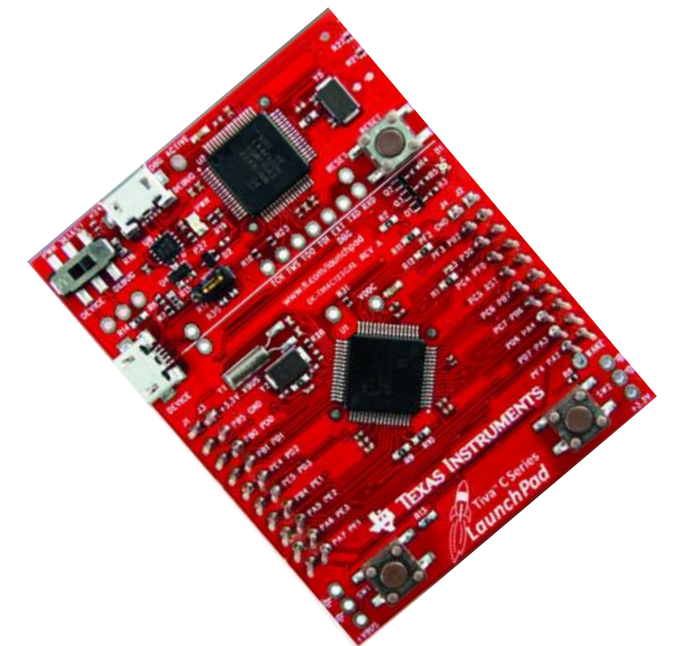
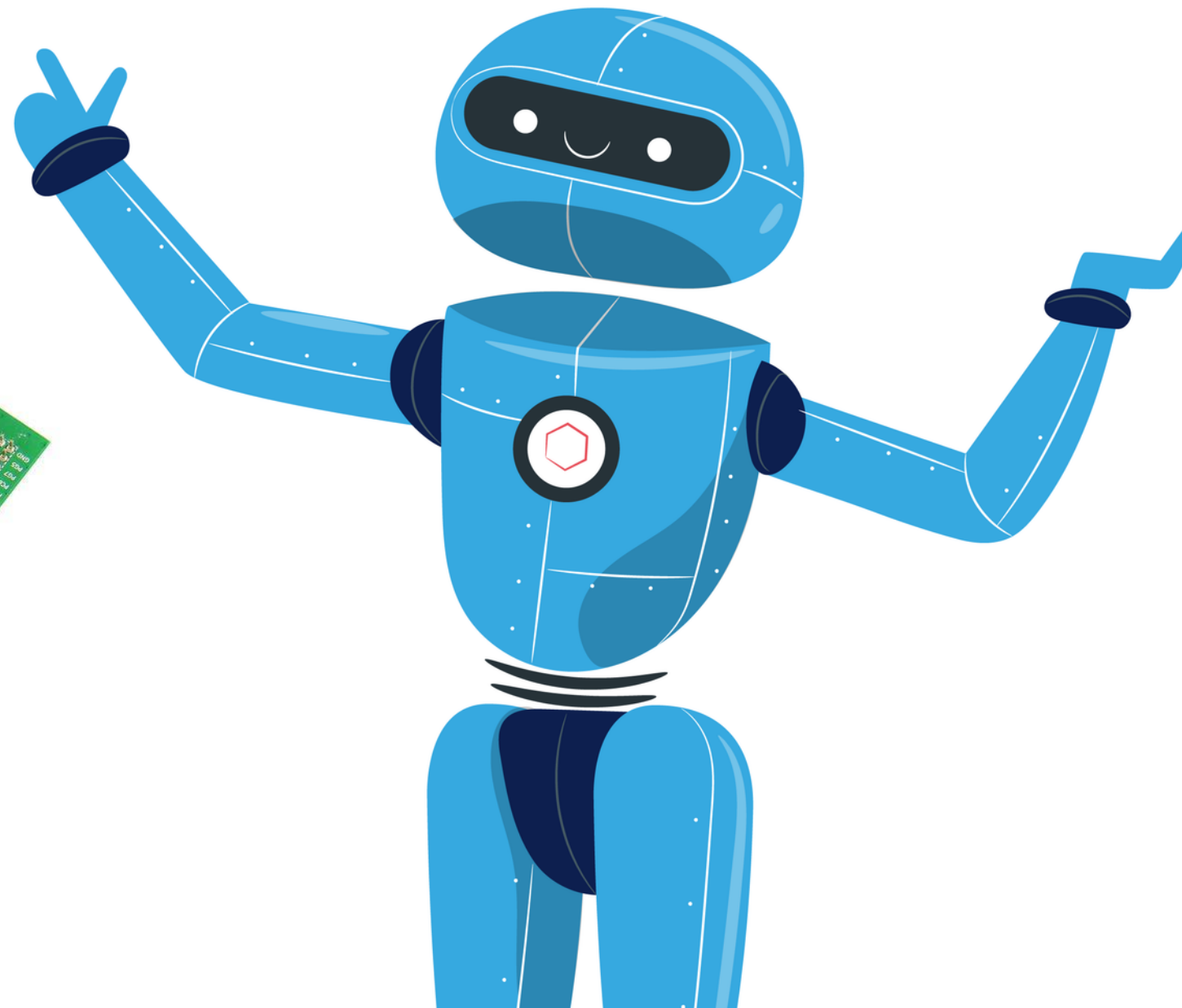
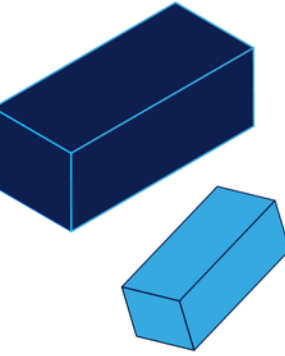


CLASE 8: TIMERS

MICROCONTROLADORES ARM



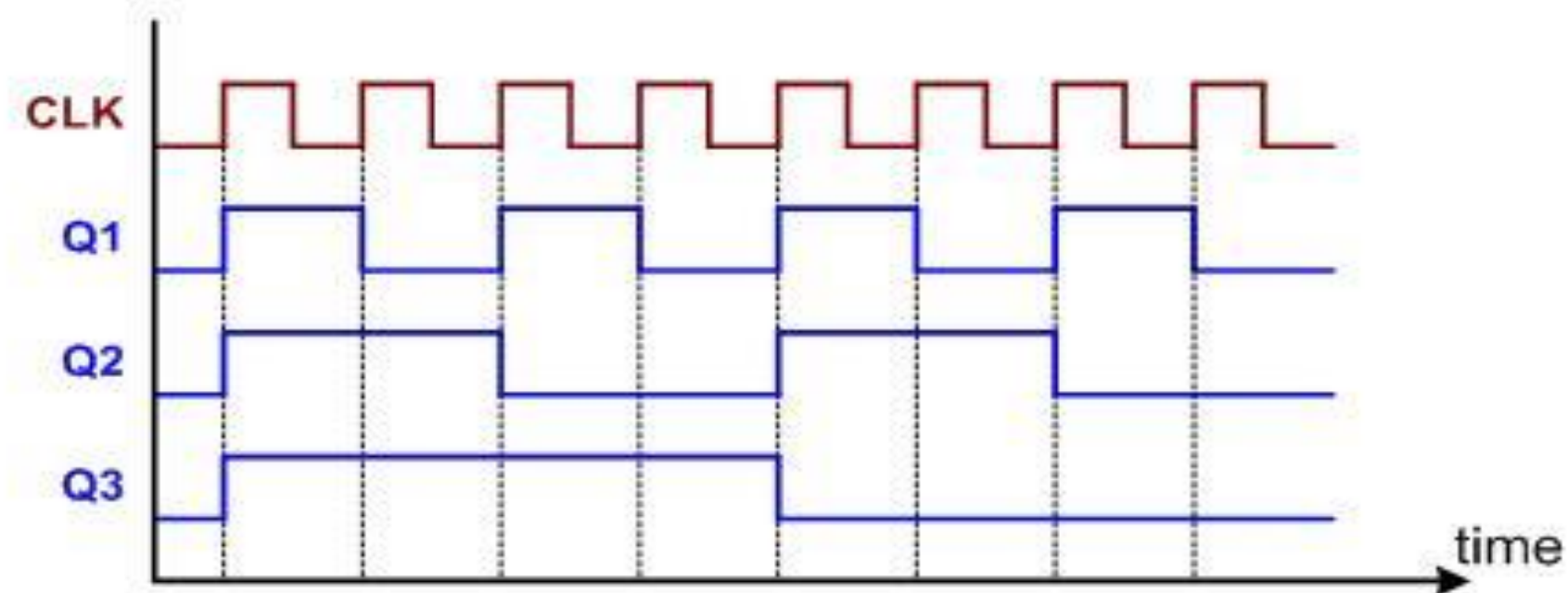
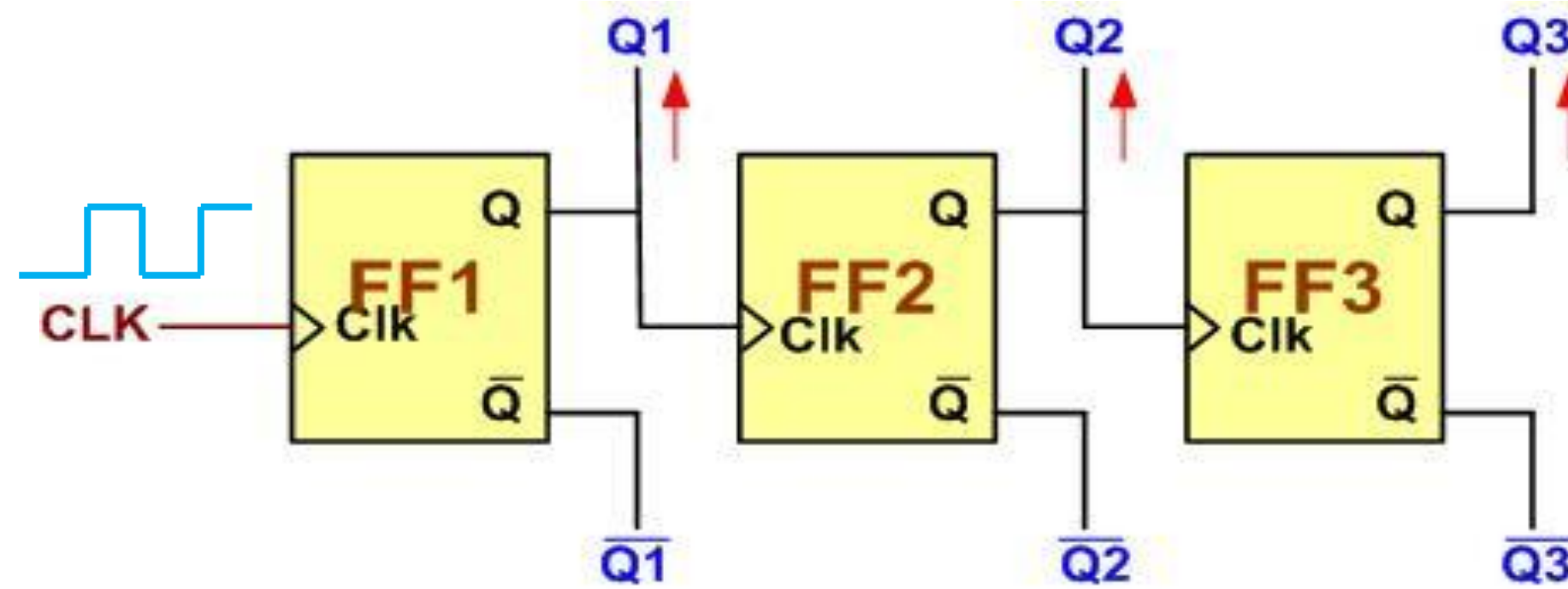


TIMERS

arm

MICRO-
CONTRO-
LADORES
ARM

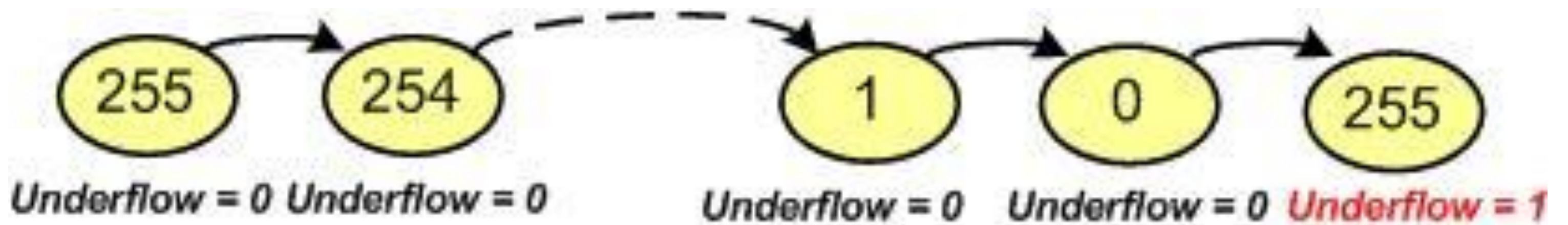
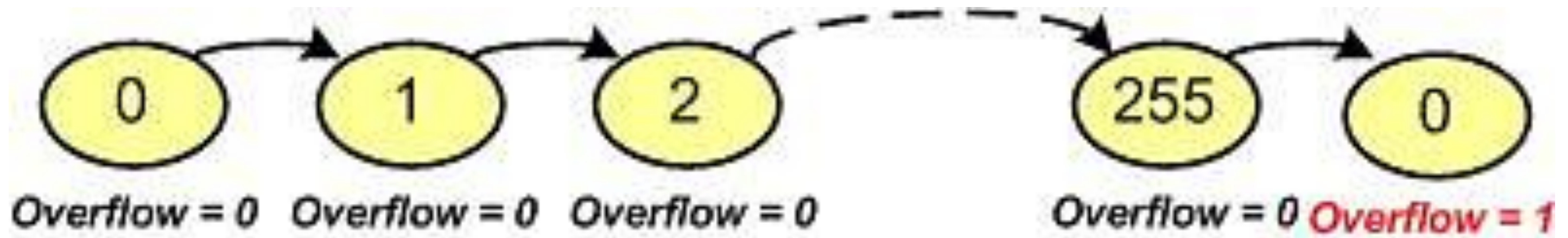
A 3-bit Counter



arm

MICRO-
CONTRO-
LADORES
ARM

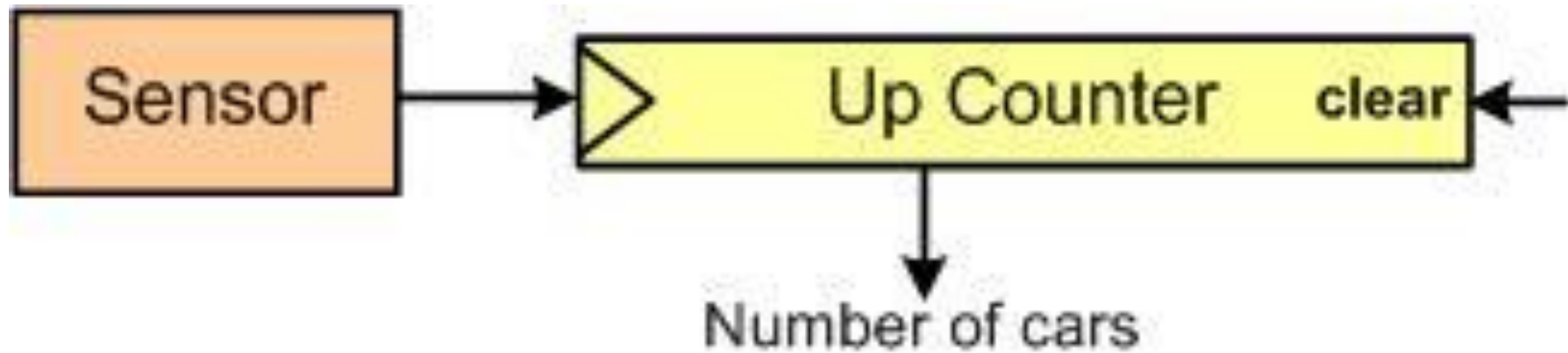
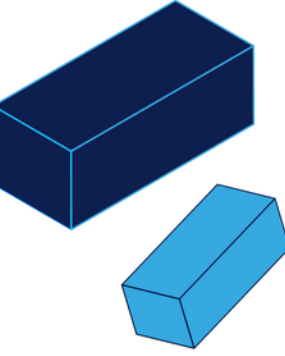
An 8-bit up/down counter stages



arm

MICRO-
CONTRO-
LADORES
ARM

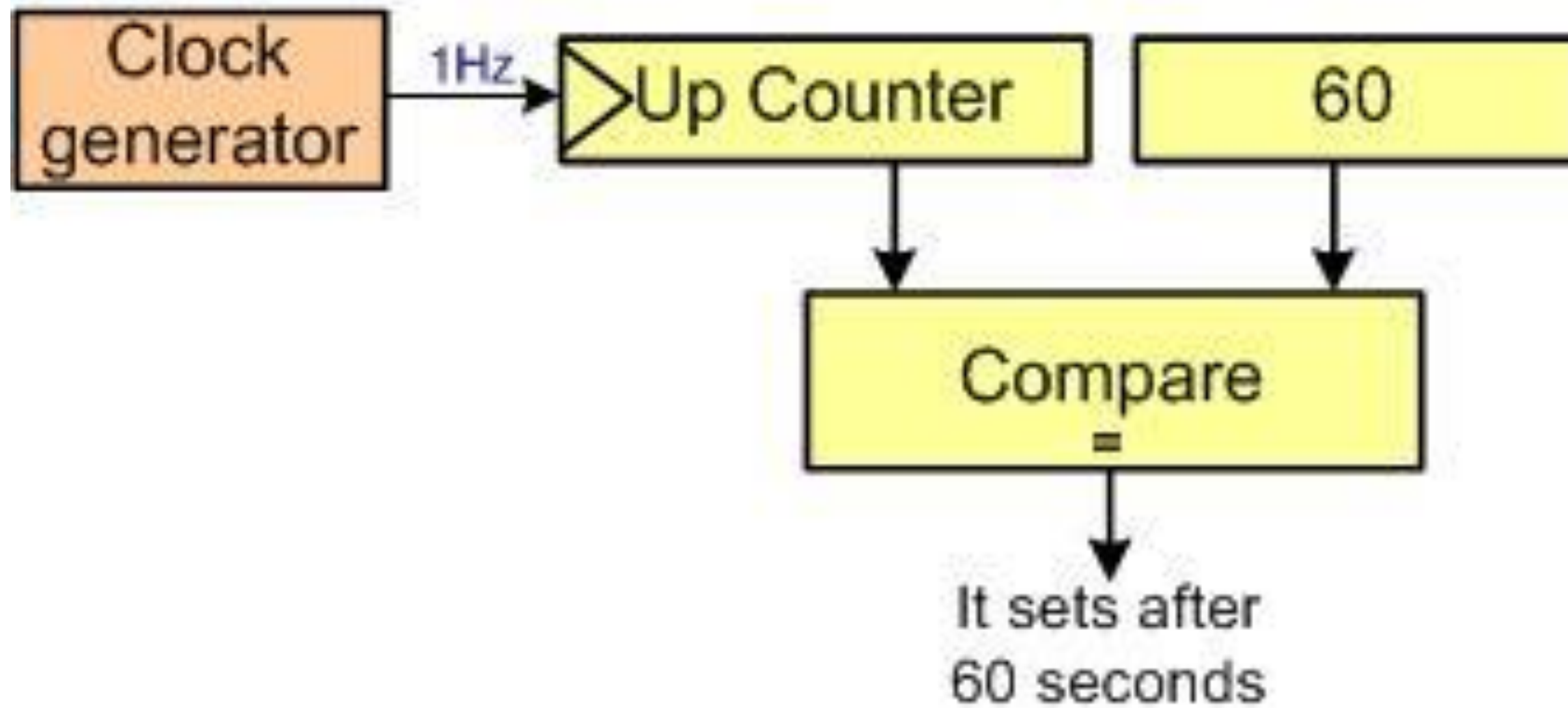
Counting Events Using a Counter



arm

MICRO-
CONTRO-
LADORES
ARM

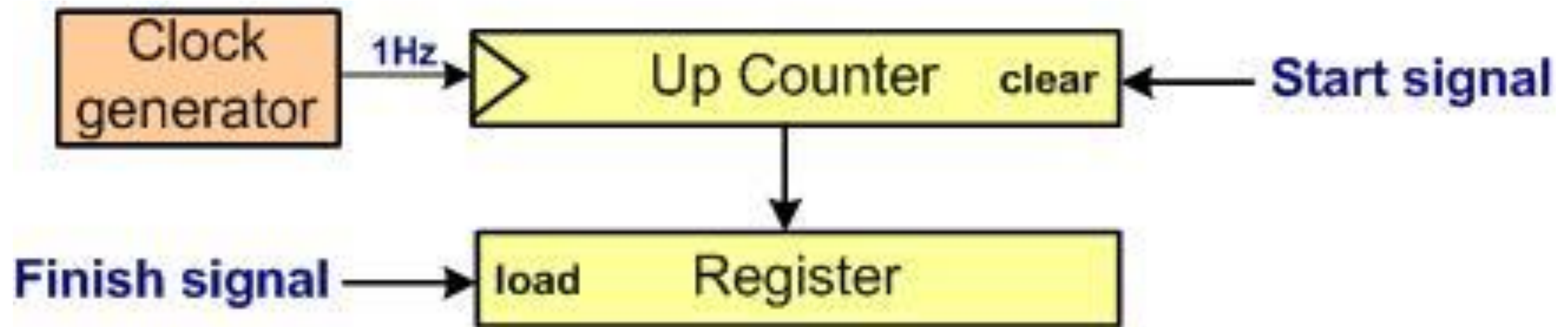
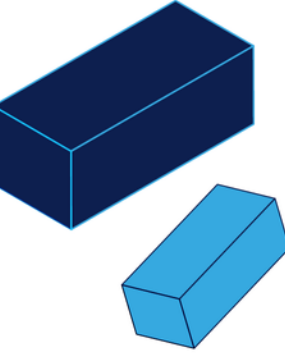
Using Counter as a Timer



arm

MICRO-
CONTRO-
LADORES
ARM

Capturing

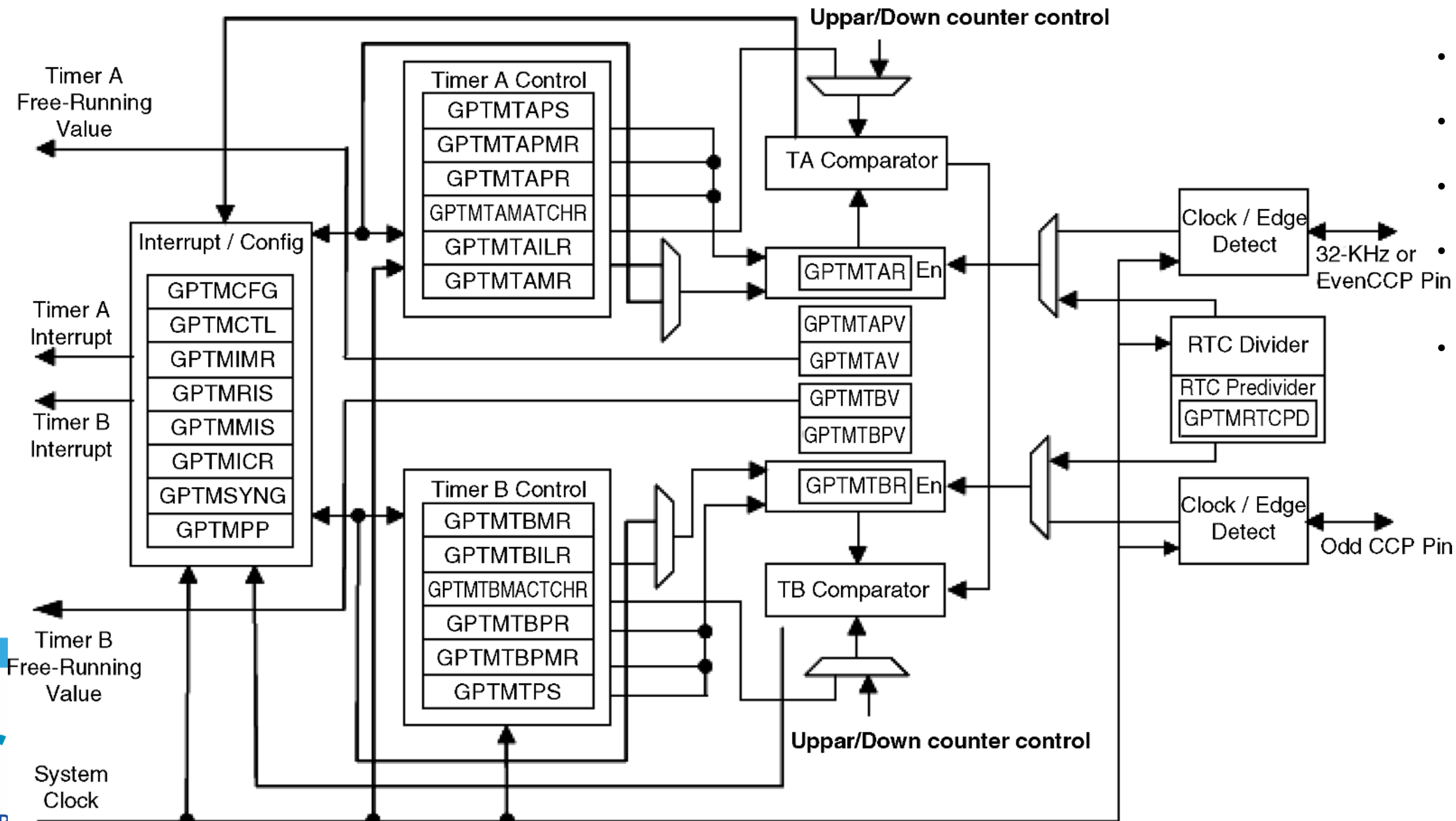


arm

MICRO-
CONTRO-
LADORES
ARM

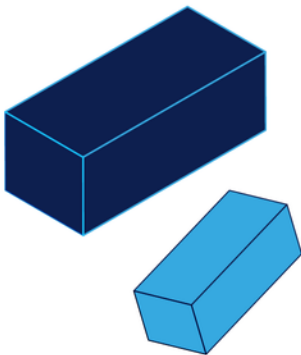
GPTM

Contiene seis bloques GPTM de 16/32 bits y seis bloques Wide GPTM de 32/64 bits



- Grupo de registro de control del temporizador A.
- Grupo de registro de control del temporizador B.
- Grupo de registro de estado del temporizador A.
- Grupo de registro de estado del temporizador B.
- Grupo de registro de interrupción y configuración de los temporizadores A y B.
- Grupo de controles externos.

GPTM



PINES DISPONIBLES

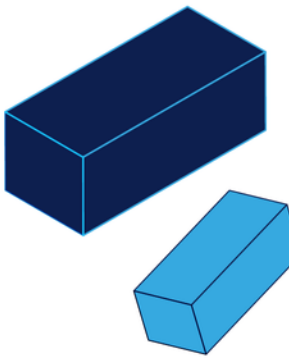
En el diagrama de bloques, los pines específicos Capture Compare PWM (CCP) disponibles dependen del dispositivo TM4C123GH6PM.

Timer	Up/Down Counter	Even CCP Pin	Odd CCP Pin
16/32-Bit Timer 0	Timer A	T0CCP0	-
	Timer B	-	T0CCP1
16/32-Bit Timer 1	Timer A	T1CCP0	-
	Timer B	-	T1CCP1
16/32-Bit Timer 2	Timer A	T2CCP0	-
	Timer B	-	T2CCP1
16/32-Bit Timer 3	Timer A	T3CCP0	-
	Timer B	-	T3CCP1
16/32-Bit Timer 4	Timer A	T4CCP0	-
	Timer B	-	T4CCP1
16/32-Bit Timer 5	Timer A	T5CCP0	-
	Timer B	-	T5CCP1
32/64-Bit Wide Timer 0	Timer A	WT0CCP0	-
	Timer B	-	WT0CCP1
32/64-Bit Wide Timer 1	Timer A	WT1CCP0	-
	Timer B	-	WT1CCP1
32/64-Bit Wide Timer 2	Timer A	WT2CCP0	-
	Timer B	-	WT2CCP1
32/64-Bit Wide Timer 3	Timer A	WT3CCP0	-
	Timer B	-	WT3CCP1
32/64-Bit Wide Timer 4	Timer A	WT4CCP0	-
	Timer B	-	WT4CCP1
32/64-Bit Wide Timer 5	Timer A	WT5CCP0	-
	Timer B	-	WT5CCP1

arm

MICRO-
CONTRO-
LADORES
ARM

GPTM



MODOS DE CONFIGURACION

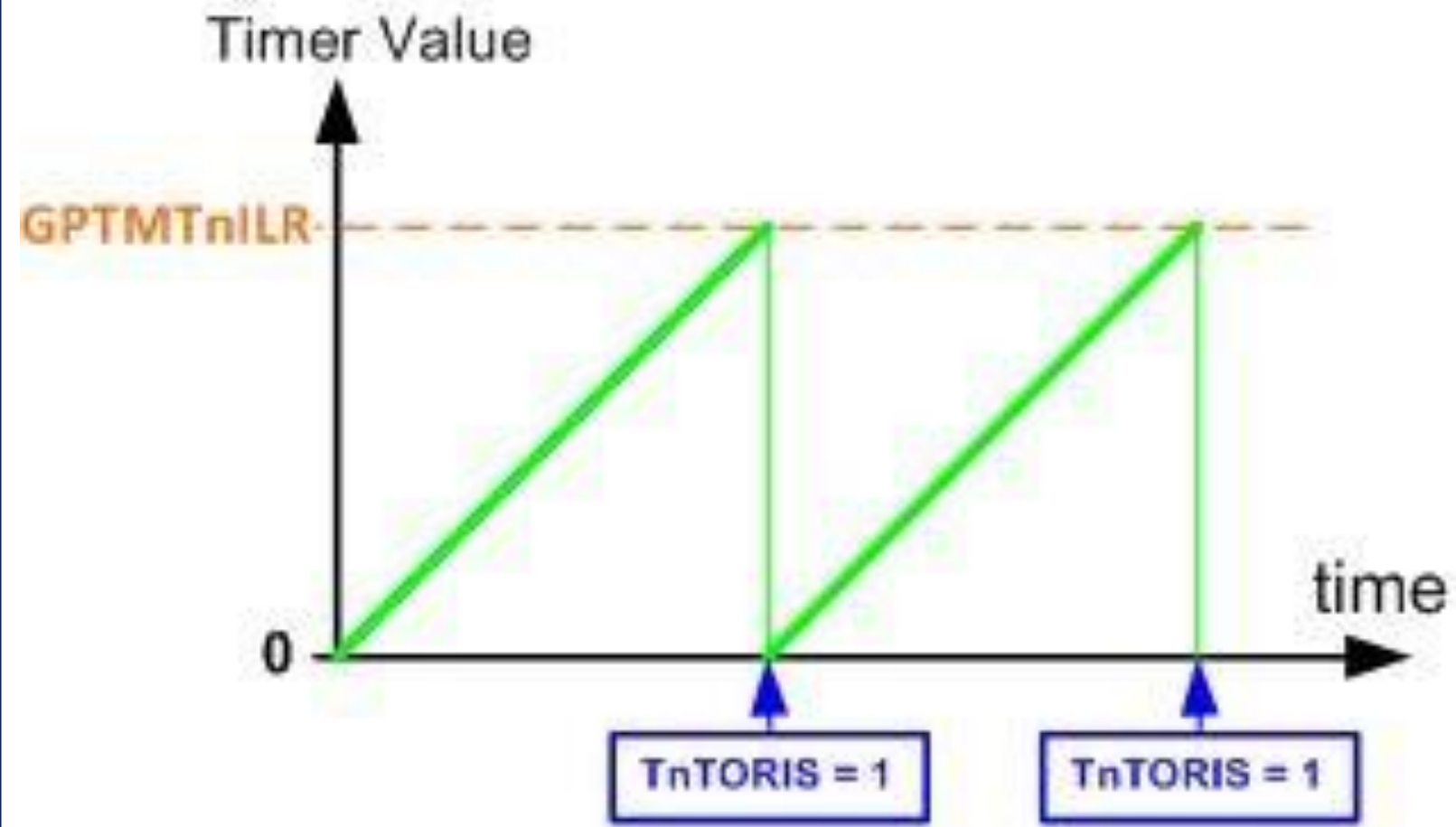
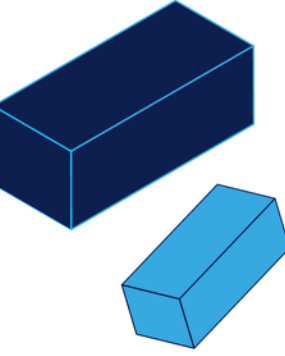
El software configura el GPTM usando el registro GPTM Configuration (**GPTMCFG**) el registro GPTM Timer A Mode (**GPTMTAMR**) y el GPTM Timer B Mode(**GPTMTBMR**). Cuando está en uno de los modos concatenados, el temporizador A y el temporizador B solo pueden funcionar en un modo. Sin embargo, cuando se configura en un modo individual, el temporizador A y el temporizador B se pueden configurar de forma independiente en cualquier combinación de los modos individuales.

Mode	Timer Use	Count Direction	Counter Size		Prescaler Size ^a		Prescaler Behavior (Count Direction)
			16/32-bit GPTM	32/64-bit Wide GPTM	16/32-bit GPTM	32/64-bit Wide GPTM	
One-shot	Individual	Up or Down	16-bit	32-bit	8-bit	16-bit	Timer Extension (Up), Prescaler (Down)
	Concatenated	Up or Down	32-bit	64-bit	-	-	N/A
Periodic	Individual	Up or Down	16-bit	32-bit	8-bit	16-bit	Timer Extension (Up), Prescaler (Down)
	Concatenated	Up or Down	32-bit	64-bit	-	-	N/A
RTC	Concatenated	Up	32-bit	64-bit	-	-	N/A
Edge Count	Individual	Up or Down	16-bit	32-bit	8-bit	16-bit	Timer Extension (Both)
Edge Time	Individual	Up or Down	16-bit	32-bit	8-bit	16-bit	Timer Extension (Both)
PWM	Individual	Down	16-bit	32-bit	8-bit	16-bit	Timer Extension

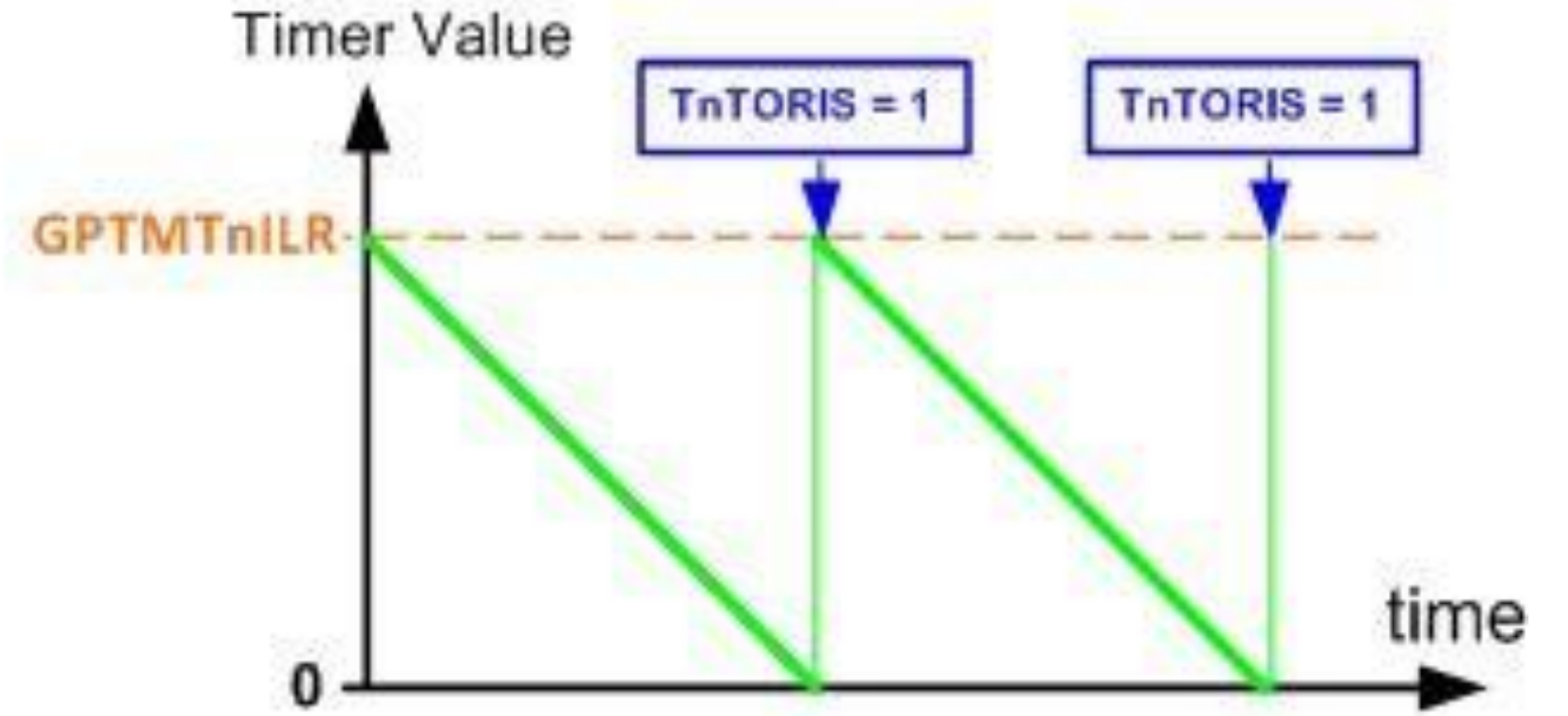
arm

MICRO-
CONTRO-
LADORES
ARM

MODOS DE CONFIGURACION



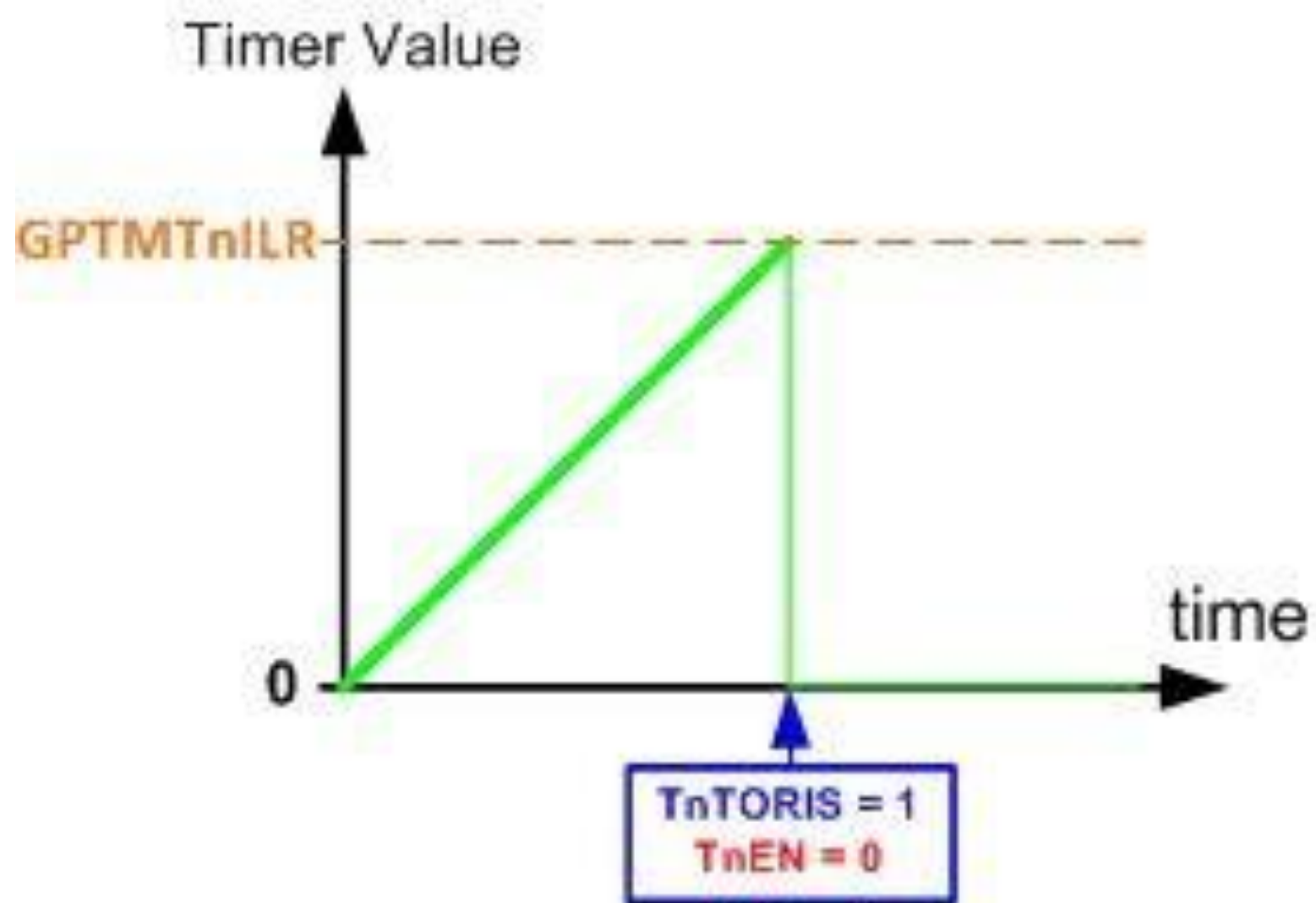
(a) up counting



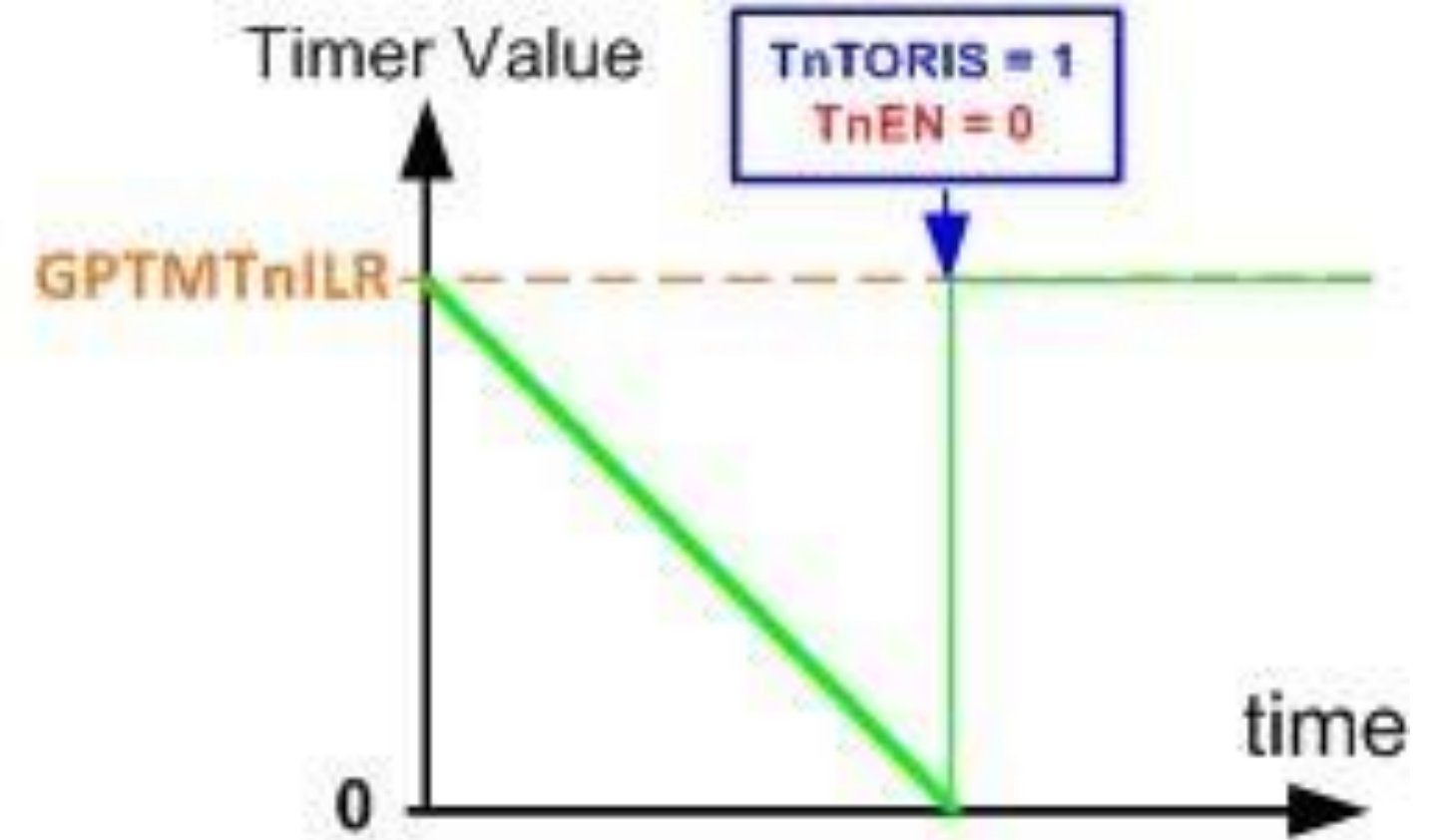
(b) down counting

GPTM

MODOS DE CONFIGURACION



(a) up counting



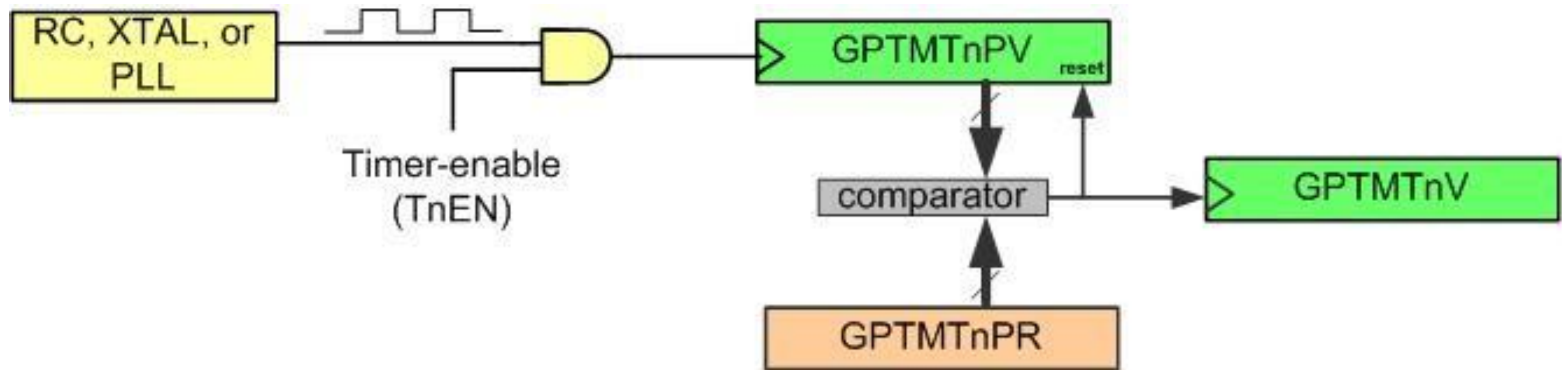
(b) down counting

arm

MICRO-
CONTRO-
LADORES
ARM

GPTM

PRESCALER

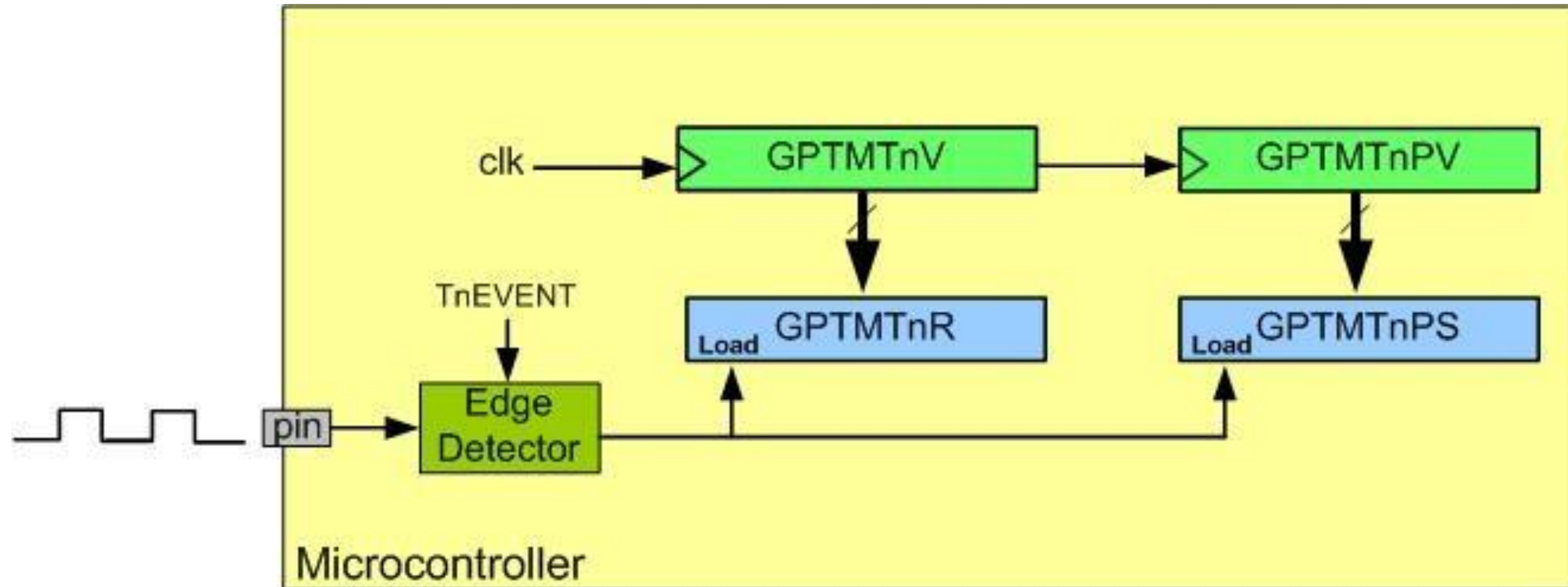


arm

MICRO-
CONTRO-
LADORES
ARM

GPTM

Input Edge Time Capturing

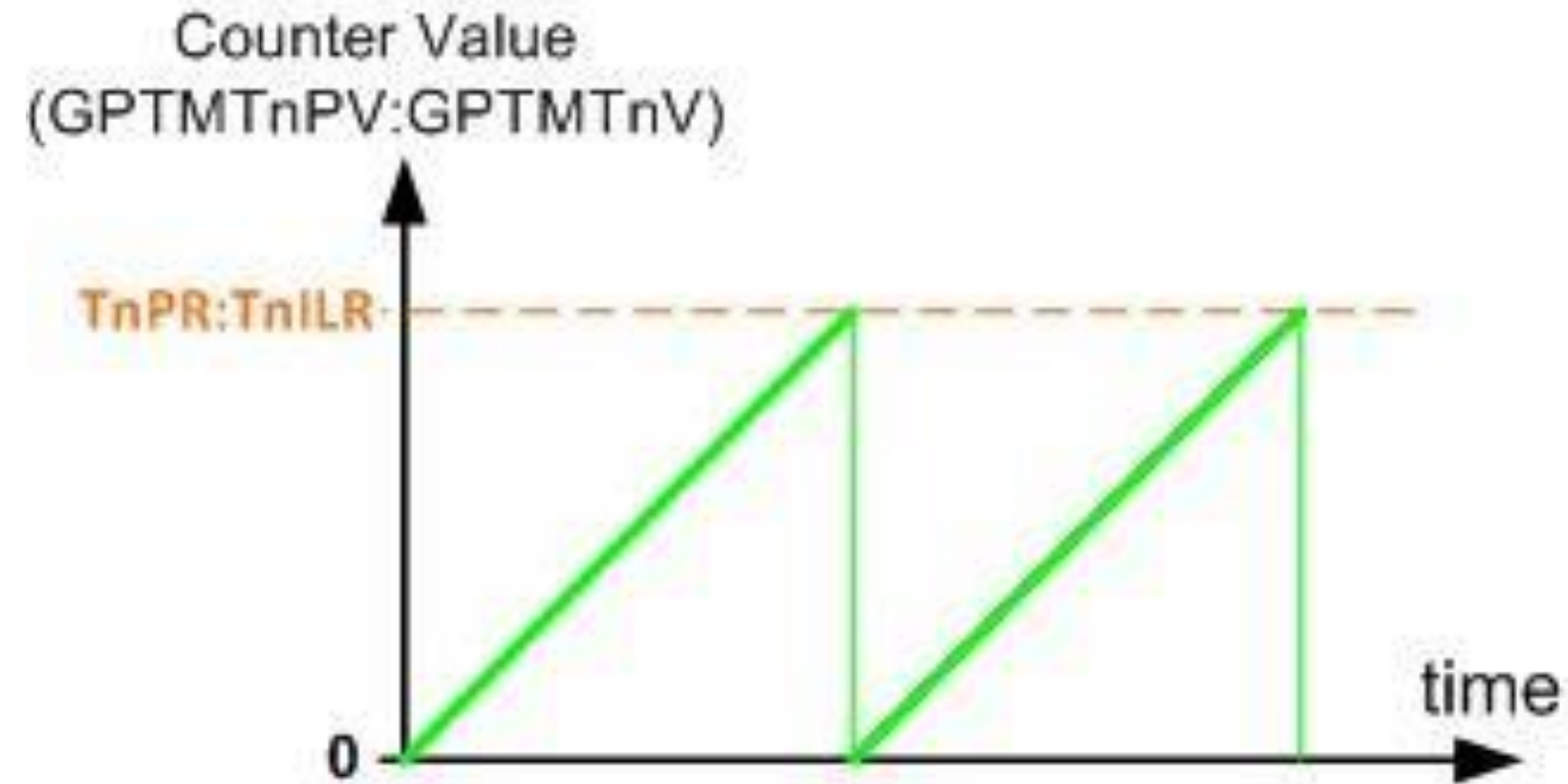
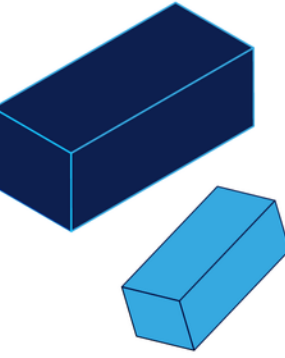


arm

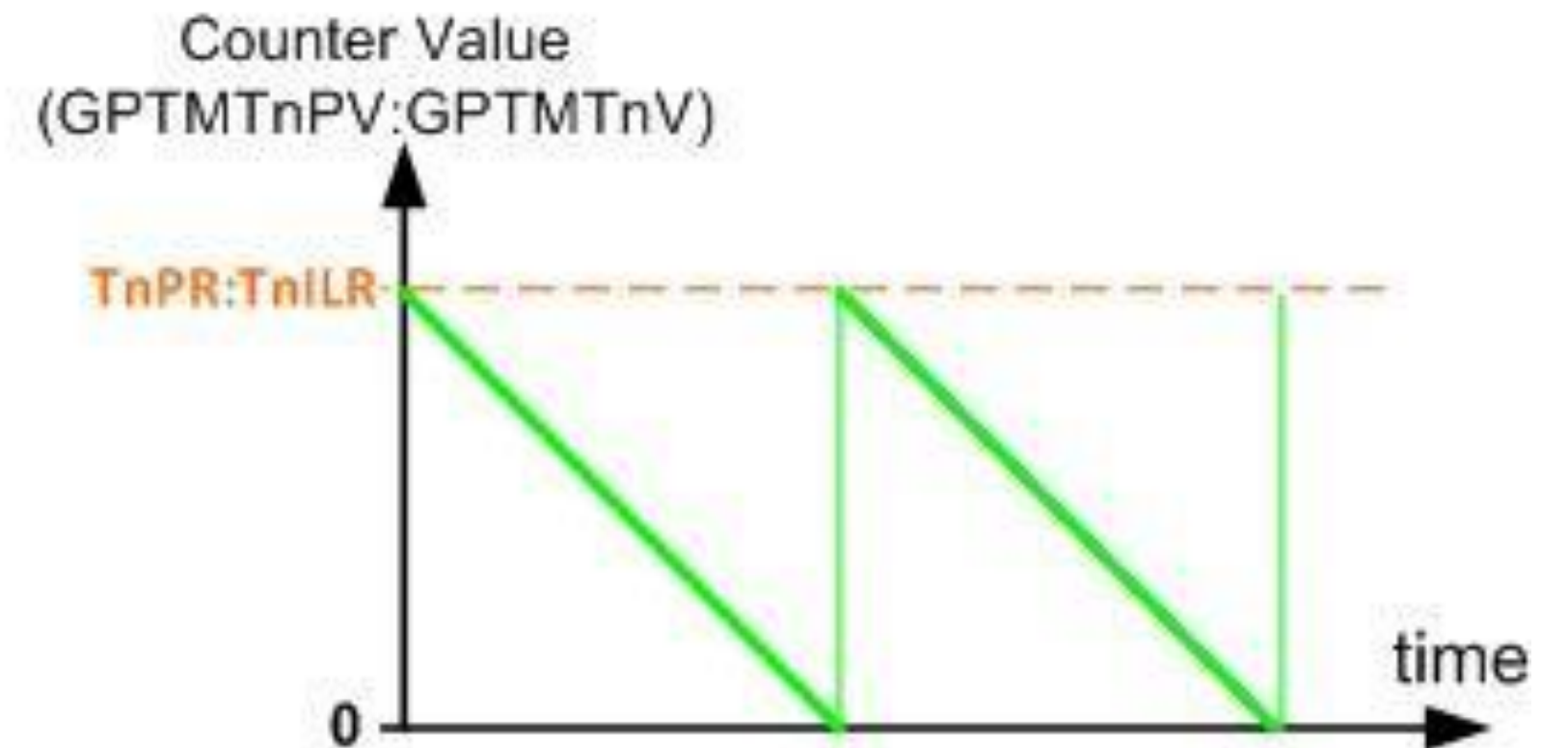
MICRO-
CONTRO-
LADORES
ARM

GPTM

Counting in Input Edge-Time Mode



(a) up counting



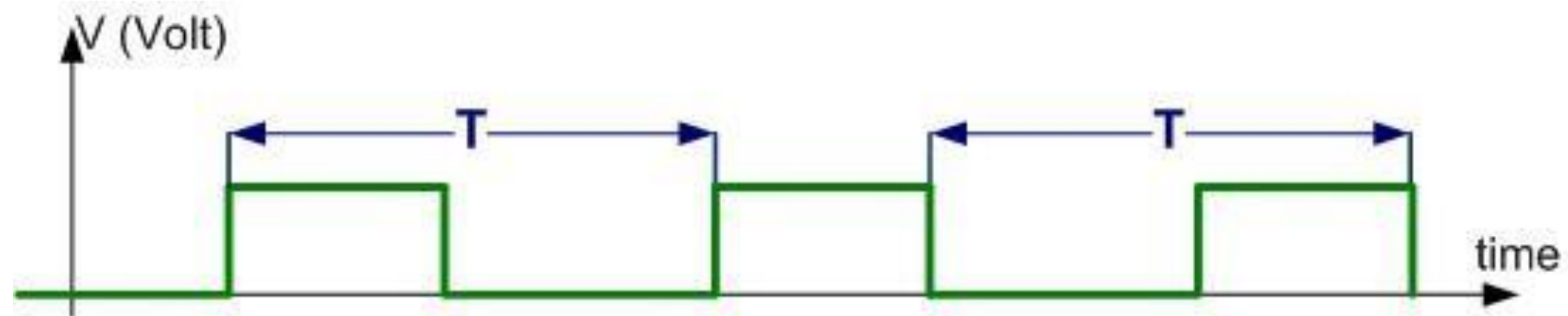
(b) down counting

arm

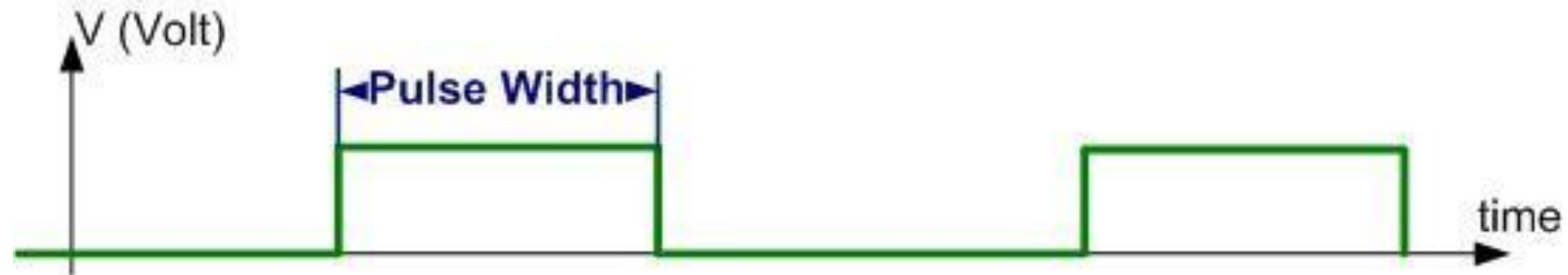
MICRO-
CONTRO-
LADORES
ARM

GPTM

Measuring Period and Pulse Width



(a) Measuring Period in Terms of the Number of Clocks Counted by the Timer



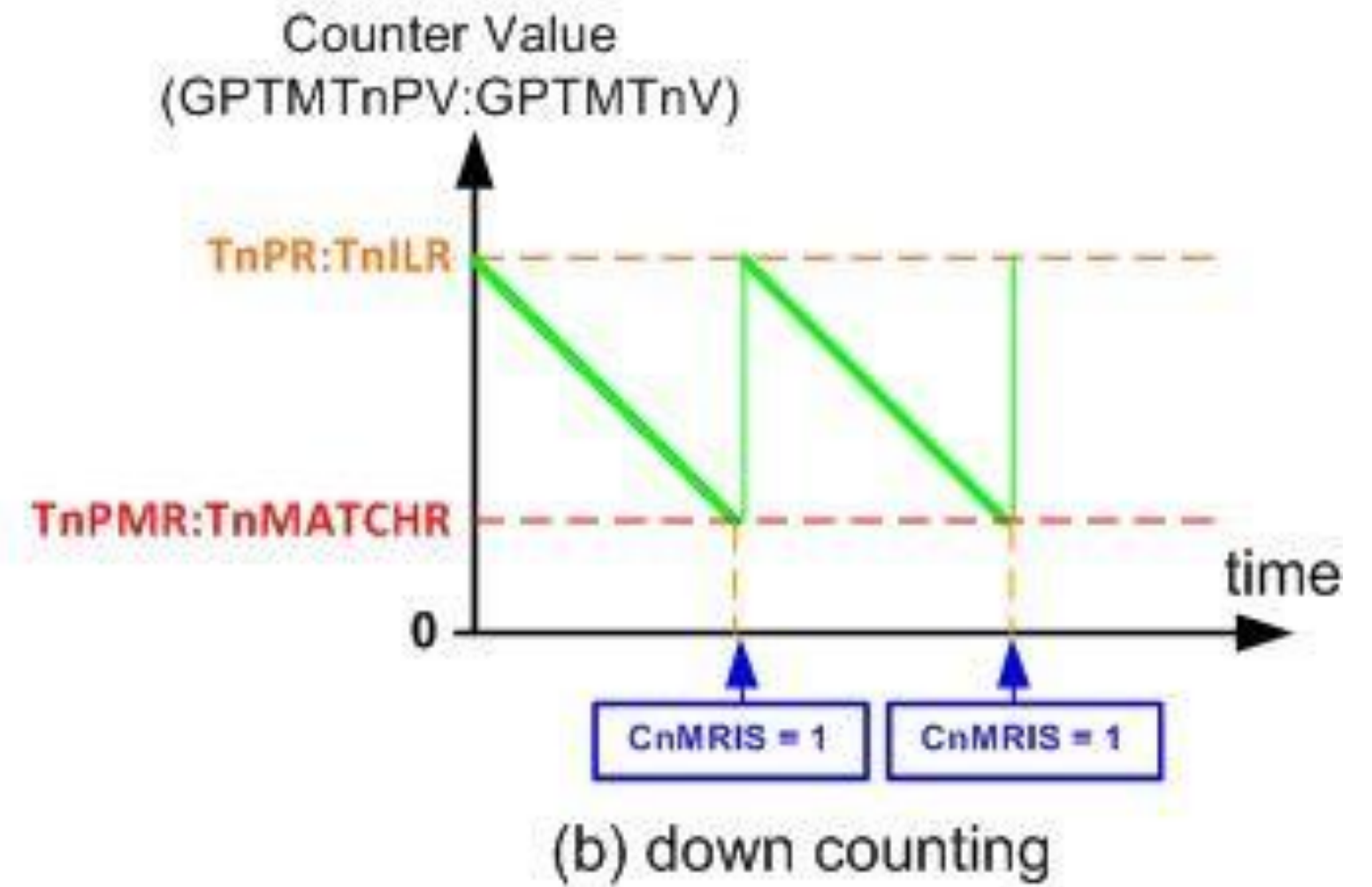
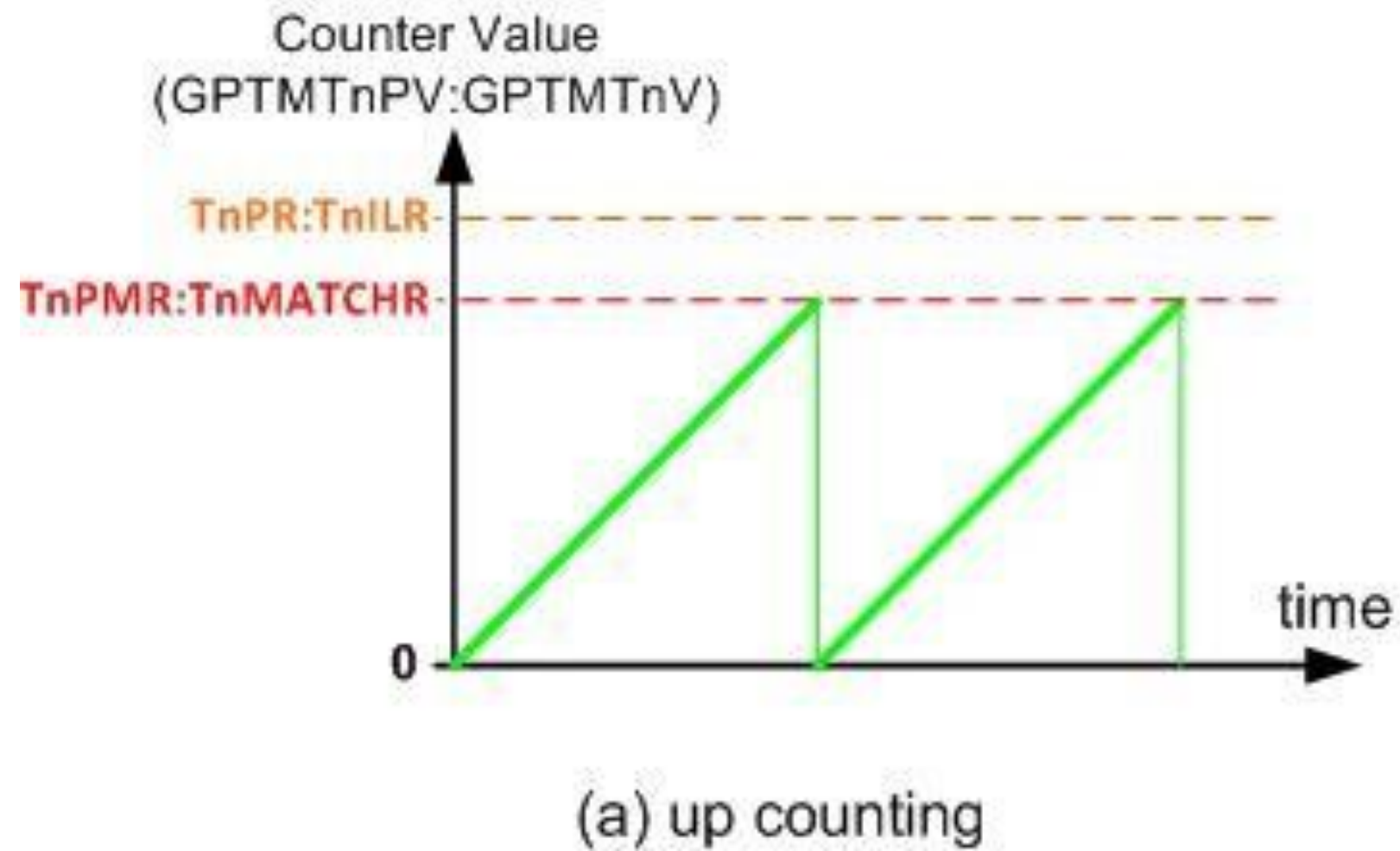
(b) Measuring Pulse Width in Terms of the Number of Clocks Counted by the Timer

arm

MICRO-
CONTRO-
LADORES
ARM

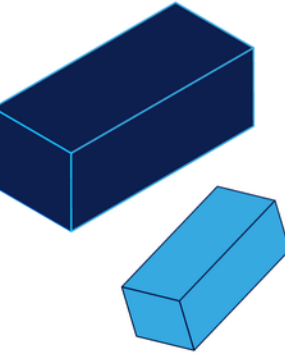
GPTM

Counting in the Input Edge-Count Mode



arm

MICRO-
CONTRO-
LADORES
ARM



EJEMPLO DE CONFIGURACION

arm

MICRO-
CONTRO-
LADORES
ARM

CONFIGURACION EN MODO PERIODICO CON INTERRUPCION

$$F_{CLOCK_CNT} = \frac{F_{CLOCK_PSC}}{PSC + 1}$$

$$UpdateEvent = \frac{F_{CLOCK_PSC}}{(PSC + 1)(ARR + 1)} \dots\dots\dots(hz)$$

$$UpdateEvent = \frac{16000000}{(15 + 1)(999 + 1)} = 1000Hz = 1kHz$$

$$CountingPeriod = \frac{(PSC + 1)(ARR + 1)}{F_{CLOCK_PSC}} \dots\dots\dots(segundos)$$

$$CountingPeriod = \frac{(15 + 1)(999 + 1)}{16000000} = 0.001 = 1ms$$

arm

MICRO-
CONTRO-
LADORES
ARM

CONFIGURACION EN MODO PERIODICO CON INTERRUPCION

```
void TIMER0A_PeriodicMode(void){
    /*0. HABILITAR EL RELOJ*/
    SYSCTL_RCGCTIMER_R |= SYSCTL_RCGCTIMER_R0;
    while(!(SYSCTL_PRTIMER_R & 1));
    /*1. DESHABILITAR EL TIMER*/
    TIMER0_CTL_R &=~ TIMER_CTL_TAEN;
    /*2. CONFIGURAR EL REGISTRO CFG*/
    TIMER0_CFG_R = 0x4;           //TIMER0 -> 16bit
    /*3. CONFIGURAR EL REGISTRO TnMR*/
    TIMER0_TAMR_R |= 0x2;         //TIMER0A-> CONTEO PERIODICO
    TIMER0_TAMR_R |= TIMER_TAMR_TACDIR; //UP COUNTER
    /*4.CONFIGURAR EL REGISTRO TnILR Y TnPR*/
    TIMER0_TAPR_R = 16-1;         //FreqCount = 16MHZ/16 = 1MHZ <-> T = 1uS
    // 1ms = X*(10^-6) -> 1000
    //50ms-> PERIODO DE CONTEO
    TIMER0_TAILR_R = 50000-1;
    /*5. CONFIGURACION DE INTERRUPCIONES*/
    TIMER0_IMR_R |= TIMER_IMR_TATOIM; //se habilita la mascara para la actualizacion de evento
    TIMER0_ICR_R |= TIMER_ICR_TATOCINT; //clear flag
    //envic
    NVIC_PRI4_R |= 1<<29;           //prioridad 2 para la interrupcion del timer0a
    NVIC_EN0_R |= 1<<19;           //se habilita la interrupcion
    /*6. habilitar el timer*/
    TIMER0_CTL_R |= TIMER_CTL_TAEN; //se habilita el conteo
    return;
}
```


CONFIGURACION EN MODO PERIODICO CON INTERRUPCION

```
void TIMER0A_Handler(void){  
    if(TIMER0_RIS_R & TIMER_RIS_TATORIS){  
        TIMER0_ICR_R |= TIMER_ICR_TATOCINT;  
        //CODE  
    }  
}
```

arm

MICRO-
CONTRO-
LADORES
ARM

CONFIGURACION EN MODO EDGE COUNT

```
void TIMER1A_edgeCountMode(void){
    /*habilitar clk*/
    SYSCTL_RCGC1_R |= SYSCTL_RCGC1_TIMER1;
    while(!(SYSCTL_PRTIMER_R & SYSCTL_PRTIMER_R1));
    SYSCTL_RCGC2_R |= SYSCTL_RCGC2_GPIOB;
    while(!(SYSCTL_PRGPIO_R & SYSCTL_PRGPIO_R1));
    /*CONFIGURACION DEL PB4 COMO SU FUNCION ALTERNATIVA*/
    GPIO_PORTB_DEN_R |= 1<<4;
    GPIO_PORTB_AFSEL_R |= 1<<4;
    GPIO_PORTB_PCTL_R |= GPIO_PCTL_PB4_T1CCP0;

    /*CONFIGURACION DEL TIMER1A*/
    TIMER1_CTL_R &=~ TIMER_CTL_TAEN;
    TIMER1_CFG_R = TIMER_CFG_16_BIT;
    TIMER1_TAMR_R |= TIMER_TAMR_TAMR_CAP; // | TIMER_TAMR_TACMR;
    TIMER1_TAMR_R |= TIMER_TAMR_TACDIR;
    TIMER1_CTL_R &=~ (0xF<<2);
    /*CONFIGURACION DEL LA INTERRUPCION*/
    TIMER1_IMR_R |= TIMER_IMR_CAEIM;
    TIMER1_ICR_R |= TIMER_ICR_CAECINT;
    //NVIC
    NVIC_PRI4_R |= 1<<13;
    NVIC_EN0_R |= 1<<21;
    /*se habilita el timer1a*/
    TIMER1_CTL_R |= TIMER_CTL_TAEN;
    return;
}
```

//timer 1 listo

//gpiob listo

//PB4 -> DIGITAL

//PB4 -> FUNCION ALTERNATIVA

//establecer funcion alternativa para TIMER1A

//TIMER1A-> 16-bit

//TIMER1A-> capture mode

//TIMER1A-> counter up

//rising edge

//Capture event

//prioridad 1

//se habilita la interrupcion

//timer1a -> enable

CONFIGURACION EN MODO EDGE COUNT

```
void TIMER1A_Handler(void){  
    if(TIMER1_RIS_R & TIMER_RIS_CAERIS){  
        TIMER1_ICR_R |= TIMER_ICR_CAECINT;  
        //CODE  
        data = TIMER1_TAR_R; //TAV  
        TIMER1_TAV_R = 0;  
    }  
}
```

arm

MICRO-
CONTRO-
LADORES
ARM

UMAKER | CENTRO DE CAPACITACIÓN
DE DESARROLLO TECNOLÓGICO